



KOREAN PATENT ABSTRACTS(KR)

Document Code:B1

(11) Publication No.1002504760000 (44) Publication Date. 20000104

(21) Application No.1019970072755 (22) Application Date. 19971223

(51) IPC Code:

G06F 3/06

(71) Applicant:

KOREA ELECTRONICS & TELECOMMUNICATIONS RESEARCH INSTITUTE

(72) Inventor:

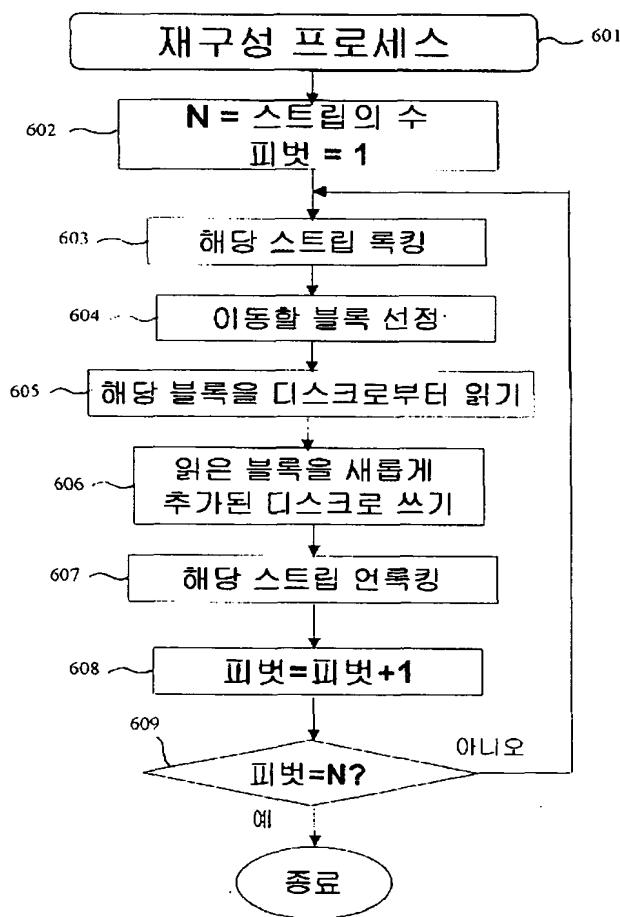
ahn, dae yeong
kim, jin pyo
kim, jung bae
lee, sang min

(30) Priority:

(54) Title of Invention

METHOD FOR REBUILDING FAST SYSTEM IN RAID LEVEL FIFTH SYSTEM

Representative drawing



(57) Abstract:

PURPOSE: The method for rebuilding the fast system in the raid level fifth system is provided to reduce the reconstitution time and the performance lowering of the system by minimizing the overhead of the system reconstitution.

CONSTITUTION: The reconfigure process takes charge of the system rebuilding(601). The number of stripe stored in the total system is N, and a variable PIVOT is 1(602). The lock flag value allocated to the relevant stripe is configured in 1(603). After setting the lock flag value, the block rearranging to the relevant stripe is selected(604). The block is read from the relevant disc(605) and written in the new disc(606). The relevant stripe is unlocked(607), 1 is

added to the PIVOT value(608), and the process(603~60) is repeatedly performed until all the stripes run is completed(60). The process perfectly ends when the rebuilding concerned to all the stripes is completed.

COPYRIGHT 2001 KIPO

if display of image is failed, press (F5)

(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) Int. Cl. ⁶ G06F 3/06	(45) 공고일자 2000년04월01일 (11) 등록번호 10-0250476 (24) 등록일자 2000년01월04일
(21) 출원번호 10-1997-0072755 (22) 출원일자 1997년12월23일	(65) 공개번호 특 1999-0053164 (43) 공개일자 1999년07월15일
(73) 특허권자 한국전자통신연구원 정선종 대전광역시 유성구 가정동 161번지 이상민	
(72) 발명자 대전광역시 서구 월평동 하나로아파트 106-504 김진표 대전광역시 서구 둔산동 은하수아파트 101-702 김중배 대전광역시 유성구 전민동 청구나래아파트 105-701 안대영 대전광역시 유성구 신성동 한울아파트 105-1101 신영무, 최승민	
(74) 대리인	

설사관 : 조형희

(54) 레이드 레벨 5 시스템에서의 빠른 시스템 재구성 방법

영세서

도면의 간단한 설명

도 1은 일반적인 RAID 시스템의 구성도.

도 2는 일반적인 RAID 레벨 5의 데이터 및 패리티 저장 방식을 나타낸 도면.

도 3은 RAID 시스템에 하나의 디스크 추가 시, 일반적인 stripe 재구성 과정을 나타낸 수행도

도 4는 본 발명에 적용된 새로운 시스템 재구성 과정을 나타낸 수행도.

도 5는 본 발명에 적용된 시스템 재구성 과정의 결과로 발생한 빈 블록을 이용한 stripe 구성도.

도 6은 본 발명에 적용된 시스템 재구성 과정의 흐름도.

〈도면의 주요 부분에 대한 부호 설명〉

101 : 호스트 시스템	102 : 어레이 제어기
103, 201 및 302 : 디스크	202 : 스트립
203 : 데이터	204 : 패리티
301 : 디스크 블록	

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 레이드(이하 RAID라 함) 레벨 5 시스템에 관한 것으로서, 특히 종래의 시스템 용량 확장을 위한 디스크 추가 시 발생하는 시스템 재구성에 따른 오버헤드를 최소화하여 긴 재구성 시간과 그로 인한 시스템의 성능 저하를 줄일 수 있는 시스템 재구성 방법에 관한 것이다.

일반적으로 RAID 레벨 5 시스템은 데이터와 패리티를 여러 디스크들로 분산 저장하고, 동시에 여러 디스크들을 접근이 가능하게 구성되므로써, 성능 향상뿐만 아니라 패리티라는 보조 정보를 제공하여 신뢰성을 향상시킨다.

따라서, 종래의 RAID 레벨 5 시스템은 시스템 용량을 확장하기 위해 디스크를 추가할 때마다, 시스템을 재구성해야 했다. 이러한 이유는 모든 데이터 및 패리티들이 분산 저장되는 구조이기 때문인데, 재구성을 위해서는 전체 디스크 블록들을 읽어 다시 새롭게 각 디스크들마다 써주어야 하기 때문이다. 그러나 이러한 방법은 읽은 블록들을 저장할 수 있는 대용량의 메모리가 필요하고, 하나의 블록들마다 한번의 읽기와 한번의 쓰기 등 모두 2번의 디스크 연산이 필요하기 때문에, 시스템 전체를 재구성하기 위해서는 아주 많은 시간이 소요되었다.

발명이 이루고자 하는 기술적 과제

따라서, 본 발명은 상술한 문제점을 해결하기 위해 시스템의 수행을 중단할 필요 없이 각 디스크의 로드(이하 load라 함)에 따라 스트립(이하 stripe이라 함) 단위로 재구성 과정을 수행한다. 이와 같은 과정은 한번의 디스크 읽기 및 쓰기를 위한 단 2번의 디스크 연산으로 처리되게 하므로, 시간을 단축시키고 시스템 성능에 미치는 영향도 적게 하는데 그 목적이 있다.

상술한 목적을 달성하기 위한 본 발명은 사용자에 의해 시스템 재구성 과정이 시작되어 전체 시스템에 저장된 스트립의 수를 N이라고 하고, 피벗 변수 값을 1로 하는 제 1 단계와, 해당 스트립이 재구성 과정을 수행 중일 때 접근 금지를 위해 해당 스트립을 롤링하고, 해당 스트립에서 재배치할 블록을 선정하여 해당 디스크로부터 그 블록을 읽어 새롭게 추가된 디스크로 쓰게 하는 제 2 단계와, 쓰기나면 해당 스트립을 풀어준 후 다음 수행을 위한 피벗 값을 증가시킨 후, 모든 스트립에 대한 수행의 완료 여부를 확인하는 제 3 단계와, 상기 확인 결과, 모든 스트립에 대한 재구성 완료 시에는 프로세스의 수행을 종료하고, 미완료 시에는 상기 제 2 단계로 복귀하여 반복된 동작을 수행하는 제 4 단계를 포함하여 이루어진 것을 특징으로 한다.

발명의 구성 및 작용

본 발명은 시스템 재구성 과정에서 필요로 되는 메모리 크기와 비용, 재구성에 걸리는 시간을 줄일 수 있도록 시스템이 가동중인 동안 디스크들에 걸리는 load에 따라 재구성 과정을 시작한다. 재구성 과정은 stripe 단위로 수행되고, 블록 재배치를 위해 필요한 디스크 연산의 수를 줄이기 위해 재배치 블록은 디스크가 종합되지 않도록 각 stripe마다 라운드 로빈(round-robin) 방식으로 선정한다. 또한 모든 stripes에 대한 재배치가 끝나면 각 디스크마다 발생되는 빈 블록들을 뒤어 새로운 stripes를 구성한다. 결과적으로 시스템 재구성에 소요되는 시간을 줄일 수 있으며, 디스크들이 idle일 때 수행하기 때문에 시스템 성능에 미치는 영향도 감소시킨다.

도 1은 일반적인 RAID 시스템의構成을 나타낸다. 어레이 제어기(Array Controller)(102)는 호스트와 디스크들을 연결하는 통로로써 호스트 시스템(101)으로부터 내려오는 입출력 요구들을 처리하고, 필요한 경우 디스크(103)들로 직접 접근하여 처리해주는 역할을 수행한다.

도 2는 일반적인 RAID 레벨 5가 지원하는 데이터 및 패리티 블록의 저장 방식을 나타낸다. N+1 RAID 레벨 5 시스템에서 각 stripe(202)은 N 개의 데이터 블록(203)과 1개의 패리티 블록(204)으로 구성되고, 각 블록은 전체 N+1 개의 디스크(201)들로 분산 저장된다. 예를 들어 Stripe 0은 Data0, Data1, Data2, Data3, 그리고 Parity0으로 구성되며, Stripe 1은 Data4, Data5, Data6, Data7, 그리고 Parity1로 구성된다. 즉, 각 블록은 Disk0부터 Disk4까지 분산 저장된다.

도 3은 RAID 레벨 5로 구성된 시스템에 새로운 디스크를 추가할 때 수행되는 시스템 재구성 방법을 나타낸 것이다.

일반적인 시스템 재구성 방식은 전체 디스크 수와 stripe 크기를 고려하여 전체 디스크 블록(301)들을 읽고, 배치 방식에 따라 새롭게 저장할 디스크(302)를 결정한 후 디스크의 처음부터 순서대로 블록들을 써주는 과정으로 처리된다.

도시된 도 3에서 Stripe 1의 Data4가 추가된 Disk5의 Data4로 이동 저장되기 때문에 나머지 Data5는 Data4의 위치로, Data6은 Data5의 위치로, Parity1은 Data6의 위치로, Data7은 Parity1의 위치로 이동된다. 따라서 하나의 stripe를 재구성하려면 각 블록마다 2번의 디스크 연산(읽기, 쓰기)을 수행해야 하기 때문에 하나의 stripe 크기가 N+1일 때 $((N+1) * 2 * T)$ 만큼의 긴 수행 시간이 소요된다. 여기서 T는 한번의 디스크 연산을 처리하는데 걸리는 시간으로 정의한다.

도 4는 본 발명에서 구현된 시스템의 재구성 방식을 나타낸 것으로, 재구성에 걸리는 소요 시간 및 시스템 성능에 미치는 오버헤드를 줄일 수 있다.

시스템을 재구성하기 위해 시스템의 수행을 중단할 필요 없이 일반 입출력 요구를 처리하면서 병행으로 재구성 과정을 처리할 수 있고, stripe 단위로 재구성을 할 때 재배치를 위해 이동해야 하는 블록의 수를 최소화할 수 있다. 재구성 과정은 각 디스크에 걸리는 load의 정도를 나타내는 높은 문턱(이하 high threshold라 함)과 낮은 문턱(이하 low threshold라 함)을 정하여 그 값에 의해 수행이 결정된다. 즉, load가 정해진 low threshold보다 낮아지는 시점에서 재구성 과정의 수행을 시작하고, 수행하는 중에 load가 high threshold보다 높아지게 되면 해당 stripe 재구성이 완료되는 시점에서 수행이 중단된다. 이 때 다음에 수행할 stripe를 알 수 있도록 피벗(이하 PIVOT이라 함)이란 변수를 사용한다. PIVOT 변수는 마지막으로 수행된 stripe의 다음 stripe을 가리키도록 하여, 재구성 과정이 다시 시작되면 이 변수가 가리키는 stripe부터 수행을 계속한다.

시스템의 재구성에 소요되는 시간을 줄이기 위해서는 재배치하는 블록의 수를 최소로 줄여야 한다. 이를 위해 본 발명에서는 stripe마다 한 블록의 이동만으로 그 수를 줄이고, 도시된 도 4와 같다. 따라서 하나의 디스크 블록에 대한 읽기와 쓰기의 2번의 디스크 연산으로 하나의 stripe를 재구성할 수 있기 때문에 전체적으로 재구성에 걸리는 시간을 상당히 감소시킬 수 있다.

도 5는 시스템 재구성 과정의 수행의 결과로 발생하는 빈 블록들을 나타낸 것이다. 재배치할 블록을 최소로 하는 방법의 결과, 빈 블록들은 서로 다른 디스크들로 분산되고, 서로 다른 디스크 위치에 저장된다. 이 빈 블록들은 시스템이 운용되는 동안 전체 시스템 용량을 넘는 데이터들을 저장해야 하는 경우, 빈 블록들을 N+1개씩 모아 새로운 stripes를 구성 사용된다.

도 6은 시스템 재구성 과정의 흐름을 나타낸 것이다.

시스템 재구성 과정은 전용 프로세스인 재구성 프로세스(Reconfigure Process)에 의해 처리된다(601). 이 프로세스는 디스크 추가 시, 사용자에 의해 수행이 시작되어 종래 시스템에 저장되어 있던 모든

stripes의 재구성이 완료되고 나면 종료된다.

전체 시스템에 저장된 stripe의 수를 N이라고 하고, Stripe 0은 재구성이 필요 없기 때문에 PIVOT 변수 값을 1로 한다(602). 우선 해당 stripe이 현재 재구성 과정을 수행 중임으로 접근을 금지한다는 것을 나타내기 위해 stripe마다 할당된 lock 플래그 값을 1로 셋팅한다(603). 플래그의 값은 해당 stripe의 재구성 과정이 완료되는 시점에서 0을 할당함으로써 해당 stripe에 대한 입출력 요구가 수행될 수 있도록 풀어준다. Lock 플래그 값을 셋팅한 후, 해당 stripe에서 재배치할 블록을 선정한다(604). 선정 방법은 이동시킬 블록의 수를 최소로 하기 위하여 Stripe 1부터 round-robin 방식으로 블록을 선정한다. 즉, 도시된 도 4와 같이, Stripe 1에서는 Data4, Stripe 2에서는 Data 9, Stripe 3에서는 Data13, 그리고 Stripe 4에서는 Data180이 선정되고, 나머지 stripe들에 대해서도 동일한 방식으로 진행된다. 블록이 선정되면 해당 디스크로부터 그 블록을 읽어 새롭게 추가된 디스크로 쓴다(605 및 606). 이 때 쓰는 위치는 디스크의 처음부터 순서대로 할당된다. 쓰기가 끝나면 해당 stripe를 풀어준 후 다음 수행을 위해 PIVOT의 값을 증가시키고, 모든 stripes에 대해 수행이 완료될 때까지 반복하여 실행한다(607, 608 및 609). 만일 실행 중에 디스크 load의 증가로 수행을 중단한다면 다음 수행에서 현재 PIVOT이 가리키는 stripe부터 재 수행한다(609). 모든 stripes에 대한 재구성이 완료되면 프로세스의 수행을 완전히 종료한다.

발명의 효과

상술한 바와 같이 본 발명에 의하면 RAID 레벨 5 시스템의 전체 시스템 용량의 확장 시, 시스템을 재구성하는데 걸리는 시간 및 그로 인해 발생하는 시스템 성능 상의 오버헤드를 감소시키는데 탁월한 효과가 있다.

(57) 청구의 범위

청구항 1

사용자에 의해 시스템 재구성 과정이 시작되어 전체 시스템에 저장된 스트립의 수를 N이라고 하고, 피벗 변수 값을 1로 하는 제 1 단계와,

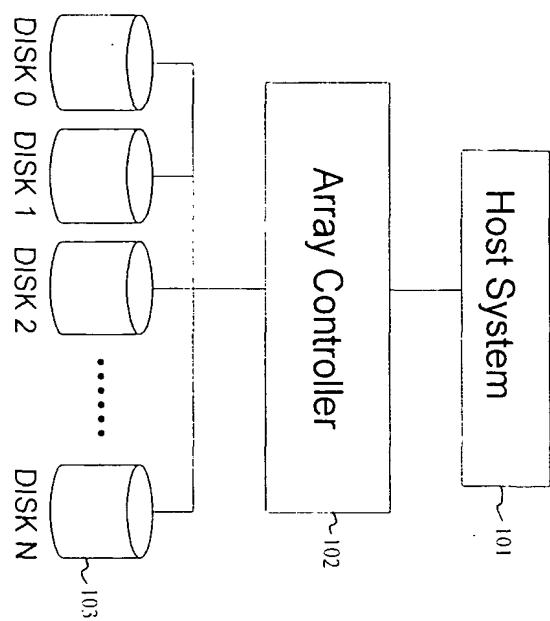
해당 스트립이 재구성 과정을 수행 중일 때 접근 금지를 위해 해당 스트립을 롤링하고, 해당 스트립에서 재배치할 블록을 선정하여 해당 디스크로부터 그 블록을 읽어 새롭게 추가된 디스크로 쓰게 하는 제 2 단계와,

쓰기가 끝나면 해당 스트립을 풀어준 후 다음 수행을 위한 피벗 값을 증가시킨 후, 모든 스트립에 대한 수행의 완료 여부를 확인하는 제 3 단계와,

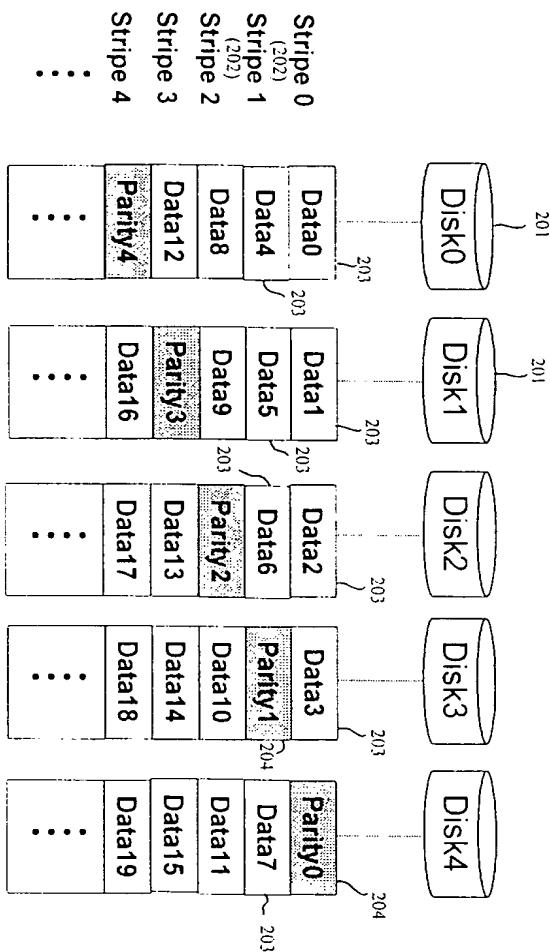
상기 확인 결과, 모든 스트립에 대한 재구성 완료 시에는 프로세스의 수행을 종료하고, 미완료 시에는 상기 제 2 단계로 복귀하여 반복된 동작을 수행하는 제 4 단계를 포함하여 이루어진 것을 특징으로 하는 레이드 레벨 5 시스템에서의 빠른 시스템 재구성 방법.

도면

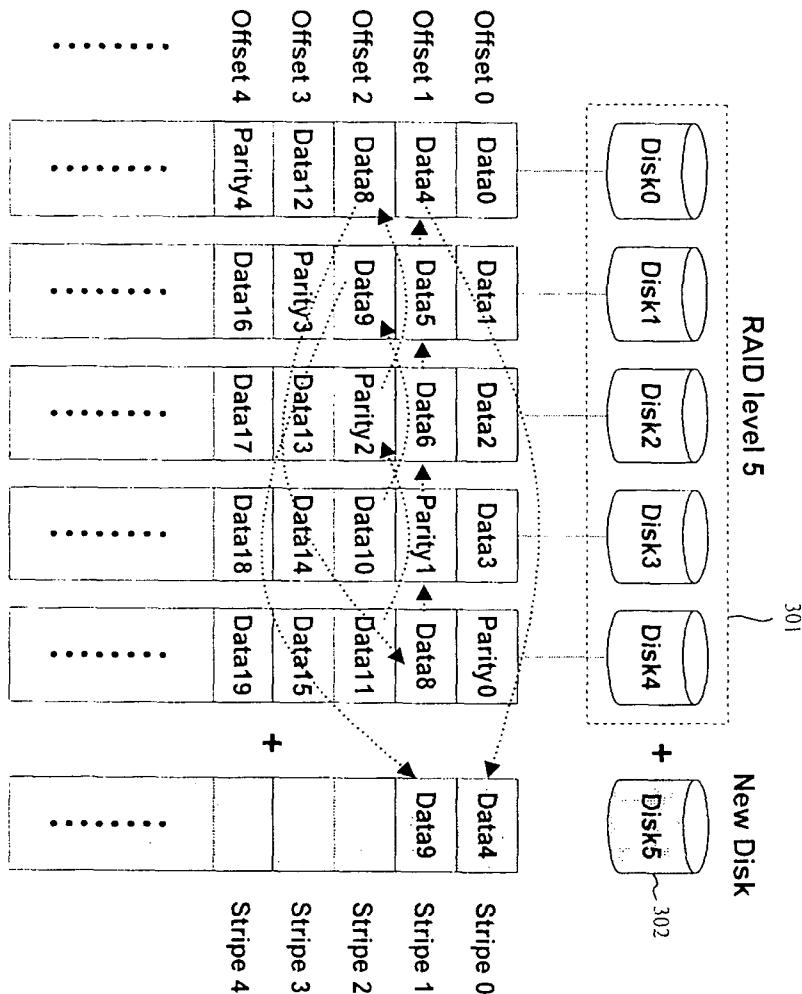
도면 1



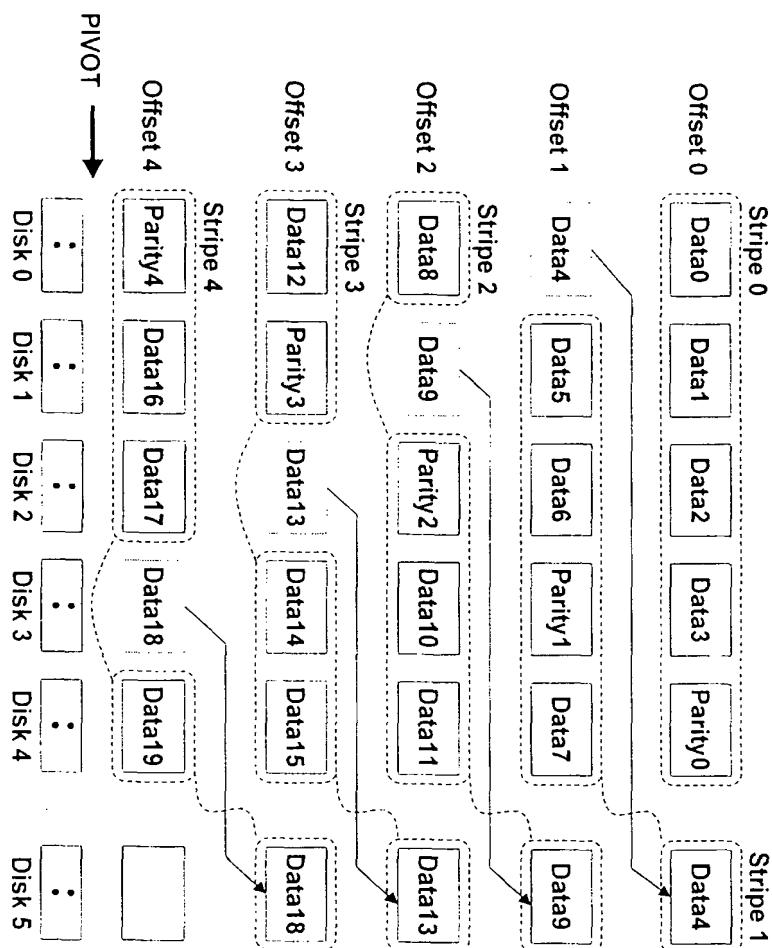
અંગુલ 2



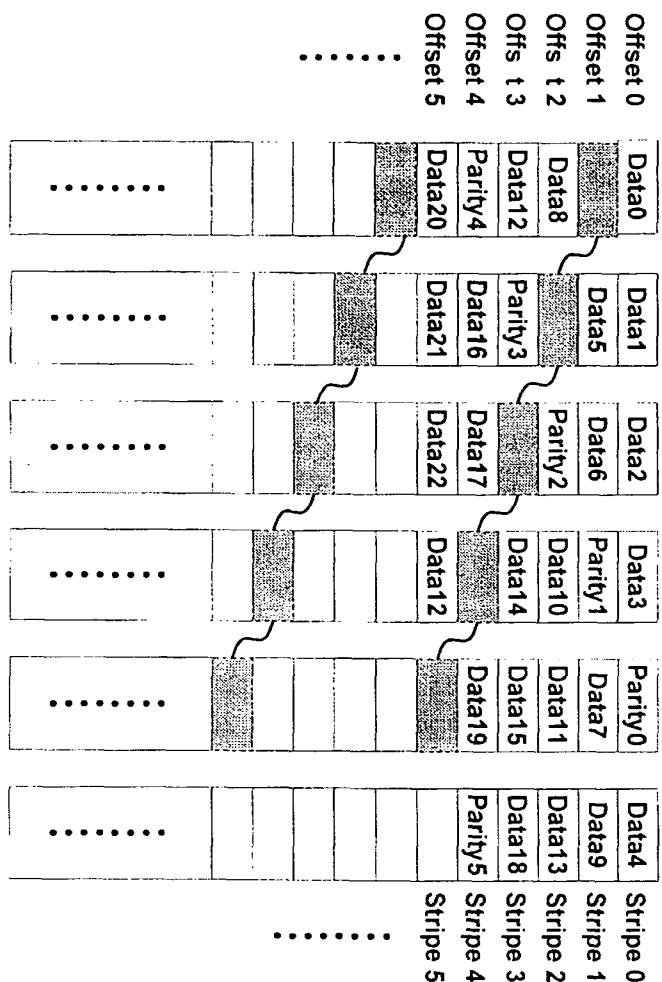
5-3



4-5



도면5



도면6

